

Research Article

Video Enhancement Using Adaptive Spatio-Temporal Connective Filter and Piecewise Mapping

Chao Wang, Li-Feng Sun, Bo Yang, Yi-Ming liu, and Shi-Qiang Yang

Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China

Correspondence should be addressed to Chao Wang, w-c05@mails.tsinghua.edu.cn

Received 28 August 2007; Accepted 3 April 2008

Recommended by Bernard Besserer

This paper presents a novel video enhancement system based on an adaptive spatio-temporal connective (ASTC) noise filter and an adaptive piecewise mapping function (APMF). For ill-exposed videos or those with much noise, we first introduce a novel local image statistic to identify impulse noise pixels, and then incorporate it into the classical bilateral filter to form ASTC, aiming to reduce the mixture of the most two common types of noises—Gaussian and impulse noises in spatial and temporal directions. After noise removal, we enhance the video contrast with APMF based on the statistical information of frame segmentation results. The experiment results demonstrate that, for diverse low-quality videos corrupted by mixed noise, underexposure, overexposure, or any mixture of the above, the proposed system can automatically produce satisfactory results.

Copyright © 2008 Chao Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. INTRODUCTION

Driven by rapid development of digital devices, camcorders and cameras are no longer used only for professional work, but step into a variety of application areas such as surveillance and home video making. While capturing videos become much easier, video defects, such as blocking, blur, noises, and contrast distortions, are often introduced by many uncontrollable factors: unprofessional video recording behaviors, information loss in video transmissions, undesirable environmental lighting, device defects, and so forth. As a result, there is an increasing demand for the technique—video enhancement, which aims at improving videos' visual qualities, while endeavoring to repress different kinds of artifacts. In this paper, we focus on two most common defects: noises and contrast distortions. While some existing software have already provided noise removal and contrast enhancement functions, it is likely that most of them introduce artifacts and could not produce desirable results for a broad variety of videos. Until now, video enhancement still remains a challenging research problem in filtering noises as well as enhancing contrast.

The natural noises in videos are quite complex; yet, fortunately, most noises can be represented using two models: additive Gaussian noise and impulse noise [1, 2].

Additive Gaussian noise generally assumes zero-mean Gaussian distribution and is usually introduced during video acquisition, while impulse noise assumes uniform or discrete distribution and is often caused by transmission errors. Thus, filters can be designed targeting the two kinds of noises. Gaussian noise can be well suppressed by bilateral filter [3], anisotropic diffusion [4], wavelet-based approaches [5], or fields of experts [6] while maintaining edges. Impulse noise filters lie on robust image statistics to distinguish noise pixels and fine features (i.e., small high-gradient regions) and often need an iterative process to reduce false detection [7–9]. Building filters for removing mixture of Gaussian and impulse noise is more practical than that for one specific type of noise with respect to natural images. The essence of mixed noise filter is to incorporate the pertinent techniques into a uniform framework that can effectively smooth the mixed noise while avoiding blurring the edges and fine features.

As to video noise removal, in addition to the above issues, temporal information should also be taken into consideration because it is more valuable than spatial information in the case of stationary scene [10]. But straightly averaging temporal corresponding pixels to smooth noise may introduce “ghosting” artifacts in the presence of camera and object motion. Such artifacts can be removed by motion compensation and a number of algorithms have been

proposed with different computational complexity [11]. However, severe impulse noise will introduce abrupt pixel changes like motions and greatly decrease the accuracy of motion compensation. Moreover, there are often not enough similar pixels for smoothing in temporal directions, owing to imperfect motion compensation or transitions between shots. Thus, a desirable video noise filter should distinguish impulse pixels and motional pixels as well as collect enough similar pixels adaptively from temporal to spatial directions.

As to contrast enhancement after noise filtering, it is quite difficult to find a universal approach for all videos owing to their diverse characteristics such as underexposed, overexposed with many fine features or with large black background. Although numerous contrast enhancement methods have been proposed, most of them are unable to automatically produce satisfactory results for different kinds of low-contrast videos, and may generate ringing artifacts in the vicinity of the edges “washed-out” artifacts [12] when having monochromatic background or noise over enhancement artifacts.

Motivated by the above observations, we propose a universal video enhancement system to automatically recover the ideal high-quality signal from noise degraded videos and enlarge their contrast to a subjectively acceptable level. For a given defective video, we introduce an adaptive spatio-temporal connective (ASTC) filter, which adapts from temporal to spatial filters based on noise level and local motion characteristics to remove mixture of Gaussian and impulse noises. Both the temporal and the spatial filters are noniterative trilateral filters, formed by introducing a novel local image statistic—neighborhood connective value (NCV) into the traditional bilateral filter. NCV represents the connective strength of a pixel to all its neighboring pixels and is a good measure for differentiating between impulse noises and fine features. After noise removal, we adopt pyramid segmentation algorithm [13] to divide a frame into several regions. Based on the areas and standard deviations of these regions, we produce a novel adaptive piecewise mapping function (APMF) to automatically enhance the video contrast. To show effectiveness of our NCV statistic, we conducted a simulation experiment by adding impulse noises into three representative pictures and reported superior noise detection performance compared with other noise filters. In addition, we tested our system on several real defective videos adding mixed noises. These videos cover diverse kinds of defectiveness: underexposure, overexposure, mixture of them, and so forth. Our outputs are much more visually pleasing than those of other state-of-art approaches.

To summarize, the contributions of this work are

- (i) a novel local image statistic for identifying impulse the pixels—*neighborhood connective value (NCV)* (Section 4),
- (ii) an *adaptive spatio-temporal connective (ASTC)* filter for reducing mixed noise (Section 5), and
- (iii) an *adaptive piecewise mapping function (APMF)* to enhance video contrast (Section 6).

In addition, Section 2 reviews previous work related to video enhancement; the system framework is represented in Section 3; Section 7 gives the experimental results, followed by conclusions in Section 8.

2. RELATED WORK

There have been much previous work on image and video noise filter and contrast enhancement. We will make a brief review on this section and describe their essential differences with our work.

2.1. Image and video noise filter

Since most natural noise can be modeled by Gaussian noise and impulse noise [1], many researchers have put great efforts on removing the two kinds of noises. Most previous Gaussian noise filters are based on anisotropic diffusion [4] or bilateral filter [3, 14, 15], both of which have similar mathematical models [16]. These methods well suppress Gaussian noise but failed to remove impulse noises owing to treating them as edges. On the other hand, most impulse noise filters are based on rank-order statistics [7, 9, 17], which perform the reordering of pixels of a 2-D neighborhood window into a 1-D sequence. Such approaches weakly exploit spatial relations between pixels. Thus, Kober et al. [8] introduced a spatially connected neighborhood (CNBH) for noise detection, which describes the connective relations of pixels with their neighborhoods, similar to our NCV statistic. But their solution only considered the pixels of CNBH, unlike ours that utilize all the neighboring pixels to characterize the structures of fine features. Furthermore, it needs to be performed iteratively to correct false detection, unlike our single-step method.

The idea of removing mixture of Gaussian and impulse noise was considered by Peng and Lucke [1] using a fuzzy filter. Then the median based SD-ROM filter was proposed [18], but it produced visually disappointing output [2]. Recently, Garnett et al. [2] brought forward an innovative impulse noise detector—rank-ordered absolute differences (ROAD)—and introduced it into bilateral filter to filter mixed noise. However, unlike our NCV approach, their approach would fail for fine feature pixels, owing to their nonoverall assumption: signal pixels should have similar intensities with at least half of their neighboring pixels.

There is a long history of research on spatio-temporal noise reduction algorithms in signal processing literature [10]. The essence of these methods is to adaptively gather enough information in temporal and spatial directions to smooth pixels while avoiding motion artifacts. Lee and Kang [19] extended anisotropic diffusion technique to the three dimensions for smoothing video noise. Unlike our approach, they did not employ motion compensation and did not treat temporal and spatial information differently. Instead, we adopt optical flow for motion estimation and use temporal filter more heavily than spatial filter. Jostschulte et al. [20] developed a video noise reduction system that used spatial and temporal filters separately while preserving edges that match a template set. The separated use of two filters limits

their performances on different kinds of videos. Bennett and McMillan [21] presented the adaptive spatio-temporal accumulation (ASTA) filter that adapts from temporal bilateral filter to spatial bilateral filter based on a tone-mapping objective and local motion characteristics. Owing to bilateral filter's limitation on removing impulse noise, their approach produces disappointing results compared with ours when applied to videos with mixed noise.

2.2. Contrast enhancement

Numerous contrast enhancement methods have been proposed such as linear or nonlinear mapping function and histogram processing techniques [22]. Most of these methods are based on global statistical information (global image histogram, etc.) or local statistical information (local histogram, pixels of neighborhood window, etc.). Goh et al. [23] adaptively used four types of fixed mapping function to process video sequences based on histogram analysis. Yet, their results heavily depend on the predefined functions, which restricts the usefulness in diverse videos. Polesel et al. [24] use unsharp masking techniques to separate image into low-frequency and high-frequency components, then amplify the high-frequency component while leaving the low-frequency component untouched. However, such methods may introduce ringing artifacts due to over enhancement in the vicinity of edges. Durand and Dorsey [25] use the bilateral filter to separate an image into details and large scale features, then map the large scale features in the log domain and leave the details untouched; thus details are more difficult to distinguish in the processed image. Recently, Chen et al. [12] brought forward the gray-level grouping technique to spread the histogram as uniformly as possible. They introduce a parameter to prevent one histogram component from occupying too many gray levels, so that their method can avoid “washed-out” artifacts, that is, over enhancing images with homochromous backgrounds. Differently, we suppress “washed-out” artifacts by disregarding the segmented regions with too small standard deviation in our mapping function forming process.

3. SYSTEM FRAMEWORK

The input to our video enhancement system is a defective video mixed with Gaussian and impulse noises and having a visually undesirable contrast. We assume that the input video V is generated by adding the Gaussian noise G and impulse noises I to a latent video L . Thus, the input video can be represented by $V = L + G + I$. Given the input defective video, the task of video enhancement system is to automatically generate an output video V' , which has visually desirable contrast and less noise. The system can be represented by a noise removal process f_2 and a contrast enhancement process f_1 as

$$V' = f_1(f_2(V)), \quad \text{where } L \approx f_2(V). \quad (1)$$

Figure 1 illustrates the system framework of our video enhancement system. Like [21], we first extract the luminance and the chrominance of each frame, and then process

the frame in luminance channel. To filter mixed noises in a given video, firstly a new local statistic—neighborhood connective value (NCV) is introduced to identify impulse noises, and then we incorporate it into the bilateral filter to form the spatial connective trilateral (SCT) filter and the temporal connective trilateral (TCT) filter. Then, we build an adaptive spatio-temporal connective (ASTC) filter adapting from TCT to SCT based on noise level and local motion characteristics. In order to deal with the presence of camera and object motion, our ASTC filter utilizes dense optical flows for motion compensation. Since typical optical flow techniques depend on robust gradient estimates and would fail on noisy low-contrast frames, we pre-enhance each frame by SCT filter and the adaptive piecewise mapping function (APMF).

In contrast enhancement procedure, we firstly separate a frame into large scale features and details using rank-ordered absolute difference (ROAD) bilateral filter [2], which preserves more fine features than other traditional filters do [26]. Then, we enhance the large scale features with APMF to achieve the desired contrast, while mapping the details using a less curved function adjusted by the local intensity standard deviation. This two pipeline method can avoid ringing artifacts even around sharp transition regions. Unlike traditional enhancement methods based on histogram statistics, we produce our adaptive piecewise mapping function (APMF) based on frame segmentation results, which provide more 2-D spatial information. Finally, the mapped large scale features, mapped details, and chrominance are combined to generate the final enhanced video. We next describe the NCV statistic, the ASTC noise filter, as well as the contrast enhancement procedure.

4. NEIGHBORHOOD CONNECTIVE VALUE

As shown in Figure 2(a), the pixels in the tiny lights are neither similar to most neighboring pixels [2] nor having small gradients in at least 4 directions [27], and thus will be misclassified as noises by [2, 27]. Comparing signal pixels in Figure 2(a) and noise pixels in Figure 2(b), we adopt the robust assumption that impulse noise pixels are always closely connected with fewer neighboring pixels than signal pixels [8]. Based on this assumption, we introduce a novel local statistic for impulse noise detection—neighborhood connective value (NCV), which measures the “connective strength” of a pixel to all the other pixels in its neighborhood window. In order to introduce NCV clearly, we need to make some important definitions first. In the following parts, let p_{xy} denotes the pixel with coordinates (x, y) in a frame, and v_{xy} denotes its intensity.

Definition 1. For two neighboring pixels p_{xy} and p_{ij} satisfying $d = |x - i| + |y - j| \leq 2$, their connective value (CV) is defined as

$$CV(p_{xy}, p_{ij}) = \alpha \times e^{-(v_{xy} - v_{ij})^2 / 2\sigma_{cv}^2}, \quad (2)$$

where α equals 1 when $d = 1$, and equals 0.5 when $d = 2$. σ_{cv} is a parameter to penalize highly different intensities and is fixed to 30 in our experiments. The CV of two

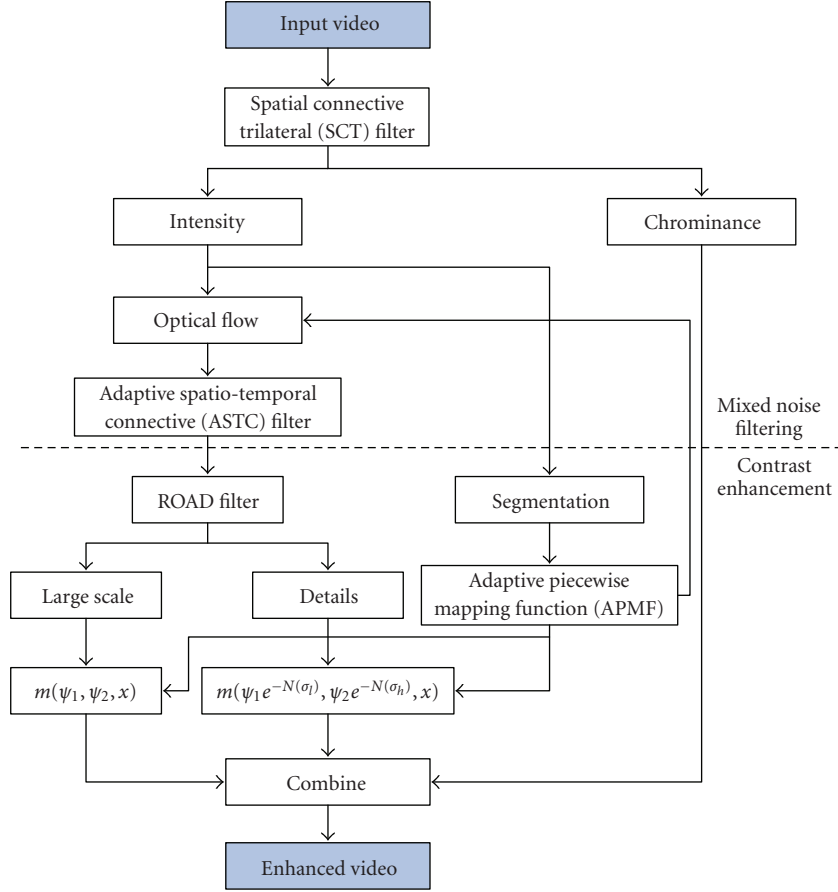


FIGURE 1: Framework of proposed universal video enhancement system, consisting of mixed noise filtering and contrast enhancement.

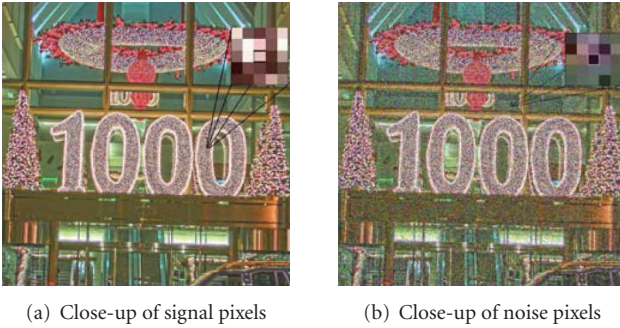


FIGURE 2: Close-ups of signal pixels in the “Neon Light” image and noise pixels in image corrupted by 15% impulse noise.

neighboring pixels assumes values in $(0, 1]$; the more similar their intensities are, the larger their CV is. CV measures the number of pixels that two neighboring pixels contribute to each other’s “connective strength.” It is perceptual rational that diagonal neighboring pixels are less closely connected than the neighboring pixels which share one identical edge, so one multiplies a factor (i.e., α) of different values to discriminate the two types of connection relationship.

Definition 2. A path P from pixel p_{xy} to pixel p_{ij} is a sequence of the pixels p_1, p_2, \dots, p_{n_p} , where $p_1 = p_{xy}$, $p_{n_p} = p_{ij}$, p_k

and p_{k+1} are neighboring pixels ($k = 1, \dots, n_p - 1$). The path connective value (PCV) is the product of CVs of all neighboring pairs along the path P

$$\text{PCV}_P(p_{xy}, p_{ij}) = \prod_{k=1}^{n_p-1} \text{CV}(p_k, p_{k+1}). \quad (3)$$

PCV describes the smoothness of a path; the more similar the intensities of pixels in the path are, the larger the path’s PCV is. PCV achieves the maximum 1 when all pixels in the path have identical intensity; thus, $\text{PCV} \in (0, 1]$. It should be noticed that there are several paths between two pixels. For example, in Figure 3, the path from p_{12} to p_{33} can be $p_{12} \rightarrow p_{22} \rightarrow p_{33}$ or $p_{12} \rightarrow p_{23} \rightarrow p_{33}$, which have PCVs of 0.0460 and 0.2497, respectively.

Although PCV well describes the smoothness of a path, it fails to give a measure for the smoothness between one pixel in the neighborhood window and the central pixel. Thus, we introduce the following definition.

Definition 3. The local connective value (LCV) of a central pixel p_{xy} with the pixel p_{ij} in its neighborhood window is the largest PCV of all the paths from p_{xy} to p_{ij}

$$\text{LCV}(p_{xy}, p_{ij}) = \max_p (\text{PCV}_p(p_{xy}, p_{ij})). \quad (4)$$

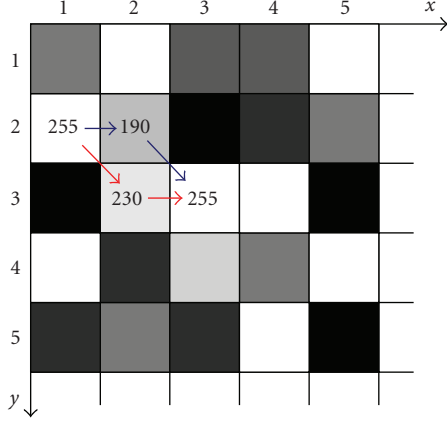


FIGURE 3: Different paths from p_{12} to p_{33} . The red path has larger PCV than the blue one does. Numbers in the figure denote the intensity values.

In the above definitions, the neighboring pixels are pixels in a $(2k + 1) \times (2k + 1)$ window, denoted by $W(p_{xy})$, with p_{xy} as the center. In our experiments, k is fixed to 2. LCV of one specific pixel equals the PCV of the smoothest path from it to the central pixel and reflects the geometric closeness and photometric similarity of it with the central one. Apparently, $LCV \in (0, 1]$.

Definition 4. The neighborhood connective value (NCV) of a pixel p_{xy} is the sum of LCVs of all its neighboring pixels

$$NCV(p_{xy}) = \sum_{p_{ij} \in W(p_{xy})} LCV(p_{xy}, p_{ij}). \quad (5)$$

NCV provides a measure of the “connective strength” of a central pixel to all its neighboring pixels. For a 5×5 neighborhood window, NCV will decrease to about 1 when the intensity of the central pixel far deviates from those of all neighboring pixels and will reach its maximum 25, when all the pixels in the neighborhood window have identical intensity, so $NCV \in (1, 25]$.

To get NCV, LCV must be calculated first. In order to compute LCV more easily, one needs to make some mathematical transform first:

$$\begin{aligned} LCV(p_{xy}, p_{ij}) &= \max_p (PCV_p(p_{xy}, p_{ij})) \\ &= \max_p \left(\prod_{k=1}^{n_p-1} CV(p_k, p_{k+1}) \right) \\ &= \exp \left(\max_p \left(\ln \left(\prod_{k=1}^{n_p-1} CV(p_k, p_{k+1}) \right) \right) \right), \end{aligned} \quad (6)$$

Let $DIS_k = \ln(1/CV(p_k, p_{k+1}))$, and one has

$$\begin{aligned} LCV(p_{xy}, p_{ij}) &= \exp \left(\max_p \left(- \sum_{k=1}^{n_p-1} DIS_k \right) \right) \\ &= \exp \left(- \min_p \left(\sum_{k=1}^{n_p-1} DIS_k \right) \right). \end{aligned} \quad (7)$$

Since $CV \in (0, 1]$, then one has $DIS_k \rightarrow 0$. Thus, one can make a graph, taking the central pixel and all its neighboring pixels as vertices and taking DIS as the cost of edge between two pixels. Therefore, the calculation of LCV can be converted to the single-source shortest path problem and can be solved by Dijkstra’s algorithm [28].

To test the effectiveness of NCV for impulse noise detection, one conducted a simulation experiments on three representative pictures: “Lena,” “Bridge,” and “Neon Light” as shown in Figure 4. “Lena” has few sharp transitions, “Bridge” has many edges, and “Neon Light” has lots of impulse-like fine features, that is, small high gradient regions. The diverse characteristics of these pictures assure the effectiveness of our experiments. Figures 5(a), 5(b), and 5(c) display quantitative results from the “Lena,” “Bridge,” and “Neon Light” images, respectively. The lower dashed lines represent the mean NCV for salt-and-pepper noise pixels—which is a discrete impulse noise model in which the noisy pixels take only the values 0 and 255—as a function of the amount of noise added, and the upper dashed line represents the mean NCV for signal pixels. The signal pixels consistently have higher mean NCVs than the impulse pixels, of which NCVs remain almost constant even with very high noise level. In contrast, the famous ROAD statistic cannot well differentiate between impulse and signal pixels in the “Neon Light” image as shown in Figure 5(d), because it assumes the signal pixels have at least half similar pixels in neighborhood window, which is coincident with the smooth regions but corrupts for fine features.

In order to enhance the NCV’s ability of noise detection, we map NCV to a new value domain and introduce the inverted NCV as

$$INCV(p_{xy}) = \frac{1}{NCV(p_{xy}) - 1} - \frac{1}{24}. \quad (8)$$

Thus, INCVs of impulse pixels will fall into large value ranges, whereas those of signal pixels will cluster near zero. Obviously, $INCV \in [0, \infty)$.

5. THE ASTC FILTER

Video is a compound of image sequences, including both spatial and temporal information. Accordingly, our ASTC video noise filter adapts from temporal to spatial noise filter. We will detail the spatial filter, the temporal filter, and the adaptive fusion strategy in this section.

5.1. The spatial connective trilateral filter

As mentioned in Section 4, NCV is a good statistic for impulse noise detection, whereas the bilateral filter [2] well suppresses Gaussian noise. Thus, we incorporate NCV into the bilateral filter to form a trilateral filter in order to remove mixed noise.

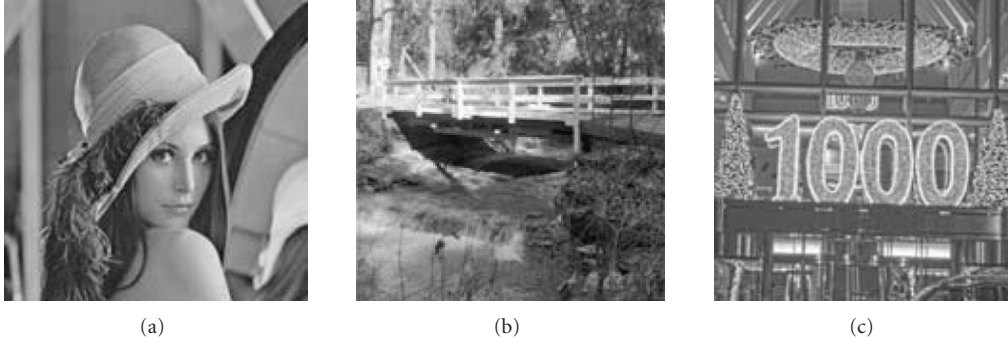


FIGURE 4: Test Images: Lena, Bridge, and Neon Light.

For a pixel p_{xy} , its new intensity v'_{xy} after bilateral filtering is computed as

$$v'_{xy} = \frac{\sum_{p_{ij} \in W(p_{xy})} \omega(p_{xy}, p_{ij}) v_{ij}}{\sum_{p_{ij} \in W(p_{xy})} \omega(p_{xy}, p_{ij})}, \quad (9)$$

$$\omega(p_{xy}, p_{ij}) = \omega_S(p_{xy}, p_{ij}) \omega_R(p_{xy}, p_{ij}), \quad (10)$$

where $\omega_S(p_{xy}, p_{ij}) = e^{-((x-i)^2 + (y-j)^2)/2\sigma_S^2}$ and $\omega_R(p_{xy}, p_{ij}) = e^{-(v_{xy}-v_{ij})^2/2\sigma_R^2}$ represent spatial and radiometric weights, respectively [2]. In our experiments, σ_S and σ_R are fixed to 2 and 30, respectively. The formula is based on the assumption that pixels locating nearer and having more similar intensities should have larger weights.

As to images with noises, intuitively, the signal pixels should have larger weights than the noise pixels. Thus, similar to the above, we introduce a third weighting function ω_I to measure the probability of a pixel being a signal pixel:

$$\omega_I(p_{xy}) = e^{-(\text{INCV}(p_{xy})^2)/2\sigma_I^2}. \quad (11)$$

Where σ_I is a parameter to penalize large INCVs and is fixed to 0.3 in our experiments. Thus, we can integrate ω_I into (10) to form a better weighting function. Yet, direct integration will fail to process impulse noise pixels because neighboring signal pixels will have lower ω_R than other impulse pixels of similar intensity. As a result, the impulse pixels remain impulse pixels. To solve this problem, Garnett et al. [2] brought forward a switch function J to determine the weight of the radiometric component in the presence of impulse noise. Similarly, our switch is defined as

$$J(p_{xy}, p_{ij}) = 1 - e^{-((\text{INCV}(p_{xy}) + \text{INCV}(p_{ij}))/2)^2/2\sigma_I^2}. \quad (12)$$

The switch J tends to reach its maximum 1, when p_{xy} or p_{ij} has large INCV, that is, with high probability of being a noise pixel; J tends to reach its minimum 0, when both p_{xy} and p_{ij} have small INCVs, that is, with high probability of being signal pixels. Thus, we introduce the switch J into (10) to control the weights of ω_R and ω_I as

$$\omega(p_{xy}, p_{ij}) = \omega_S(p_{xy}, p_{ij}) \omega_R(p_{xy}, p_{ij})^{1-J(p_{xy}, p_{ij})} \times \omega_I(p_{ij})^{J(p_{xy}, p_{ij})}. \quad (13)$$

According to the new weighting function, for impulse noise pixels, ω_R is almost “shut off” by the switch J , while ω_I and ω_S work to remove the large outliers; for other pixels, ω_I is almost “shut off” by the switch J , and only ω_R and ω_S work to smooth small amplitude noise without blurring edges. Consequently, we build the spatial trilateral connective (SCT) filter by merging (9) and (13).

Figure 6 shows the outputs of ROAD and SCT filters for the “Neon Light” image corrupted by mixed noise. ROAD filter is based on a rank-order statistic for impulse detector and the bilateral filter. It can well smooth the mixed noise with PSNR = 23.35 but blur lots of fine features such as the tiny lights in Figure 6(b). In contrast, our SCT filter preserves more fine features and produces more visually pleasing output with PSNR = 24.13, as shown in Figure 6(c).

5.2. Trilateral filtering in time

As to videos, temporal filtering is more important than spatial filtering [10], but irregular camera and object motions often degrade the performance. Thus, robust motion compensation is quite necessary. Optical flow is a classical approach for this problem; however, it depends on robust gradient estimation and will fail for noisy, underexposed, or overexposed images. Therefore, we pre-enhance the frames with SCT filter and our adaptive piecewise mapping function, which will be detailed in Section 6. Then, we adopt the `cvCalcOpticalFlowLK()` function of the intel open source computer vision library (Opencv) to compute dense optical flows for robust motion estimation. Too small and too large motions are deleted; also, half-wave rectification and Gaussian smoothing are applied to eliminate noises in optical flow field [29].

After motion compensation, we adopt the similar approach to SCT filter in temporal direction. In temporal connective trilateral (TCT) filter, we define the neighborhood window of a pixel p_{xyt} as $W(p_{xyt})$, which is a $(2m+1)$ -length window in temporal direction with p_{xyt} as the middle. In our experiments, m is fixed to 10. Noticing that the pixels in the window may have different horizontal and vertical coordinates in frames, but they are on the same tracking

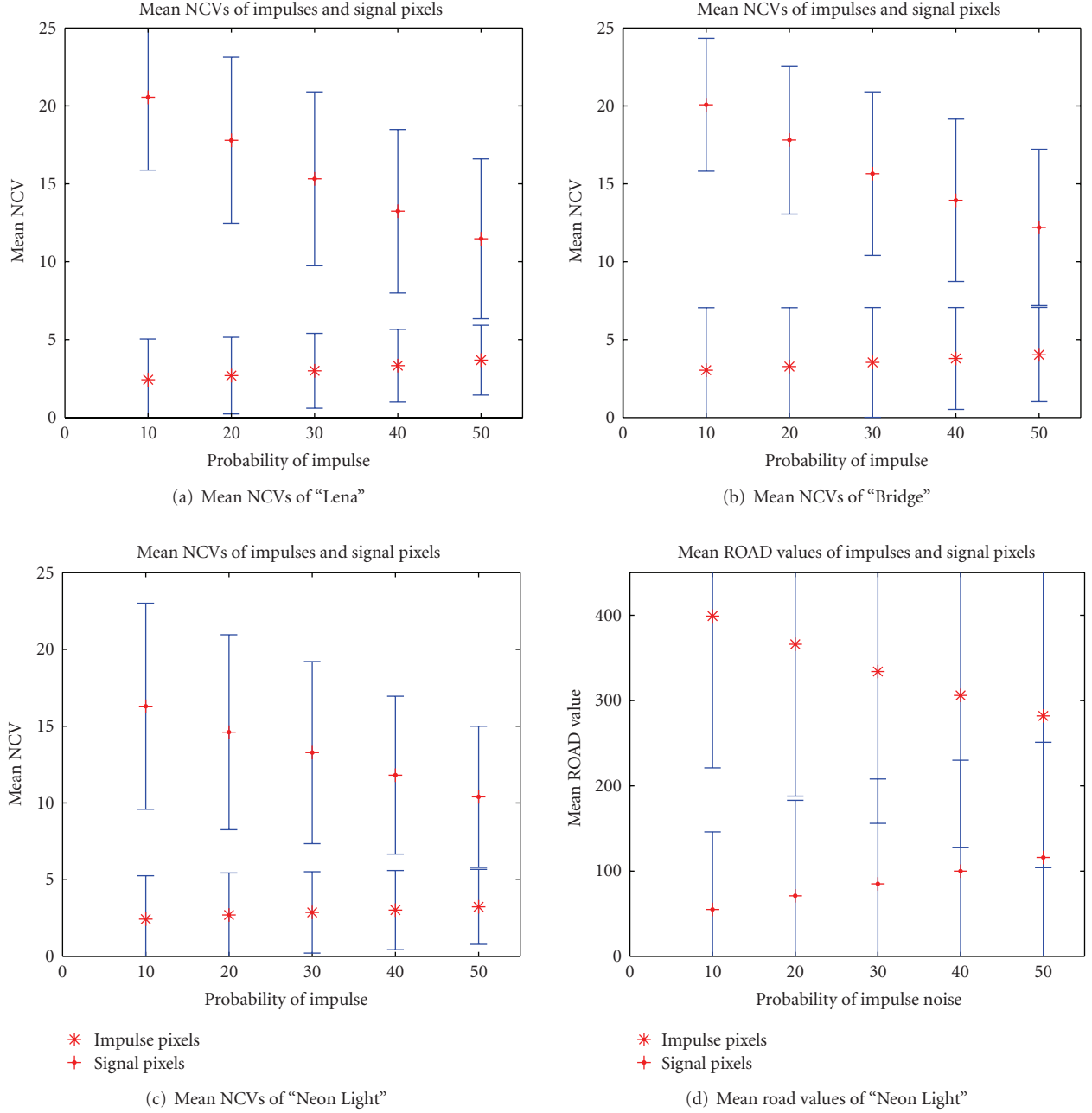


FIGURE 5: The mean NCV as a function of the impulse noise probability of signal pixels (cross points) and impulse pixels (star points) in the (a) "Lena" image, (b) "Bridge" image, and (c) "Neon Light" image, with standard deviation error bars indicating the significance of the difference; (d) the mean ROAD values of impulse pixels (star points) and signal pixels (cross points) with standard deviation error bars.

path generated by the optical flow. Thus, the TCT filter is computed as

$$v'_{xyt} = \frac{\sum_{p_{ijk} \in W(p_{xyt})} \omega(p_{xyt}, p_{ijk}) v_{ijk}}{\sum_{p_{ijk} \in W(p_{xyt})} \omega(p_{xyt}, p_{ijk})},$$

$$\omega(p_{xyt}, p_{ijk}) = \omega_S(p_{xyt}, p_{ijk}) \omega_R(p_{xyt}, p_{ijk})^{1-J(p_{xyt}, p_{ijk})} \quad (14)$$

$$\times \omega_I(p_{ijk})^{J(p_{xyt}, p_{ijk})},$$

where $\omega_S(p_{xyt}, p_{ijk}) = e^{-((x-i)^2 + (y-j)^2 + (t-k)^2)/2\sigma_S^2}$ and $\omega_R(p_{xyt}, p_{ijk}) = e^{-(v_{xyt} - v_{ijk})^2/2\sigma_R^2}$. ω_I and J are defined the same as (11) and (12), respectively.

The TCT filter can well differentiate impulse noise pixels from motional pixels and smooth the former while leaving the later almost untouched. For impulse noise pixels, the switch function J in TCT filter will "shut off" the radiometric component and the spatial weight is used to smooth them; for motional pixels, J will "shut off" the impulsive component and TCT filter reverts to bilateral filter,

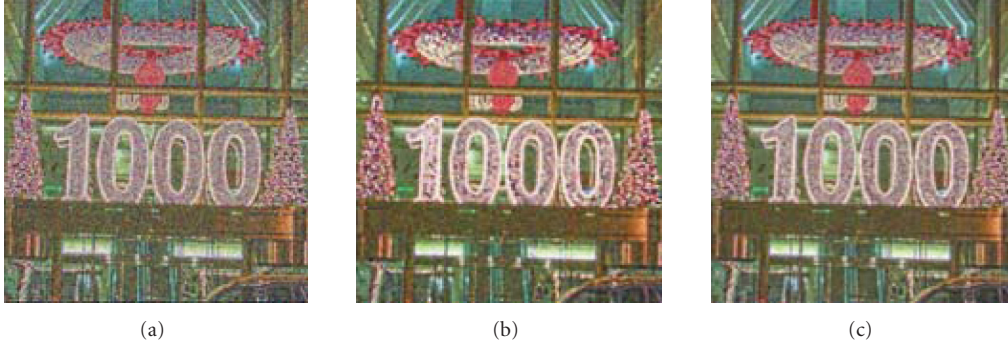


FIGURE 6: Comparing ROAD filter with our SCT filter on image corrupted by mixed Gaussian ($\sigma = 10$) and impulse noise (15%). (a) Test image, (b) result of ROAD filter (PSNR = 23.35), and (c) result of SCT filter (PSNR = 24.13).

which takes the motional pixels as “temporal edges” and leaves them unchanged.

5.3. Implementing ASTC

Although TCT filter is based on robust motion estimation, there are often not enough similar pixels in temporal direction for smoothing in presence of complex motions. As a result, the TCT filter fails to achieve desirable smoothing results and have to convert to spatial direction. Thus, a threshold is necessary to determine whether a sufficient number of temporal similar pixels are gathered; this threshold then can be used as a switch between temporal and spatial filters (in [21]), or as a parameter adjusting importance of the two filters (in our ASTC). If the threshold is too high, then for severely noisy videos, there are always not enough valuable temporal pixels, and temporal filter becomes useless; if the threshold is too low, then no matter how noisy a video is, the output will be always based on unreliable temporal pixels. Accordingly, we introduce an adaptive threshold η like [21], but further considering local noise levels:

$$\eta = \kappa \times \lambda_{xy} = \frac{1}{25} \sum_{p_{ij} \in W(p_{xy})} e^{-(\text{INCV}(p_{ij})^2)/2\sigma_t^2} \times \lambda_{xy}. \quad (15)$$

In the above formula, κ presents the local noise level and is computed in a spatial 5×5 neighborhood window. κ reaches its maximum 1 in good frames and decreases with the increase of noise level. λ_{xy} is the gain factor of current pixel and equals the tone mapping scales in our adaptive piecewise mapping function, which will be detailed in Section 6. Thus, the more mapping scale is and less noises exist, the larger η becomes; the less mapping scale is and more noises exist, the smaller η becomes. Such characteristics assure the threshold working well for different kinds of videos.

Since the temporal filter outperforms the spatial filter when gathering enough temporal information, we propose the following criteria for the fusion of temporal filter and spatial filter.

- (1) If a sufficient number of temporal pixels are gathered, only temporal filter is used.

- (2) On the other hand, even if temporal pixels are insufficient, the temporal filter should still more dominant over the spatial one in the fused spatio-temporal filter.

Based on these two criteria, we propose our adaptive spatio-temporal connective (ASTC) filter, which adaptively fuses the spatial connective trilateral filter and temporal connective trilateral filter as

$$\begin{aligned} \text{ASTC}(p_{xyt}) = & \text{thr}\left(\frac{w_t}{\eta}\right) \times \text{TCT}(p_{xyt}) + \left(1 - \text{thr}\left(\frac{w_t}{\eta}\right)\right) \\ & \times \text{SCT}(p_{xyt}), \end{aligned} \quad (16)$$

where

$$\begin{aligned} \text{thr}(x) = & \begin{cases} 1 & \text{if } x > 1, \\ x & \text{otherwise,} \end{cases} \\ w_t = & \sum_{p_{ijk} \in W(p_{xyt})} w(p_{xyt}, p_{ijk}), \end{aligned} \quad (17)$$

which represents the sum of pixel weights in temporal direction. If $w_t > \eta$ (i.e., sufficient temporal pixels), $\text{thr}(w_t/\eta) = 1$, then ASTC filter regresses to temporal connective trilateral filter; if $w_t \leq \eta$ (i.e., insufficient temporal pixels), $\text{thr}(w_t/\eta) < 1$, ASTC filter will use the temporal connective trilateral filter to gather pixels in temporal direction first, and then use the spatial connective trilateral filter to gather the remaining number of pixels in spatial direction.

6. ADAPTIVE PIECEWISE MAPPING FUNCTION

We have described the process of filtering mixture of Gaussian and impulse noises from defective videos. However, contrast enhancement is another key issue. In this section, we will show how to build the tone mapping function as well as how to automatically adjust important parameters and smooth the function in time.

6.1. Generating APMF

As the target of our video enhancement system is to deal with diverse videos, our tone mapping function needs to work

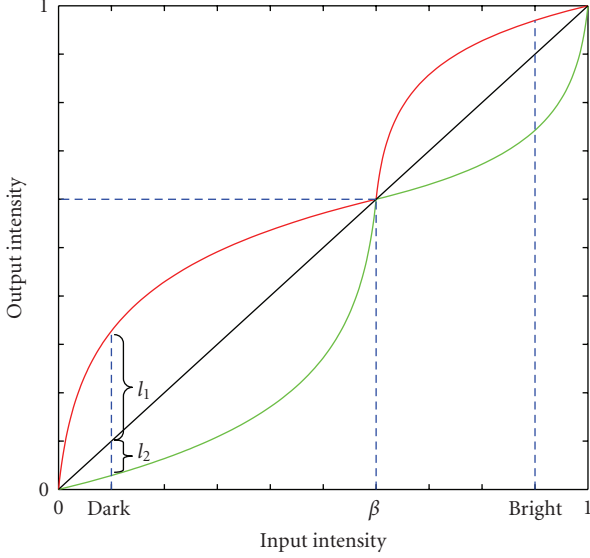


FIGURE 7: Our adaptive piecewise mapping function. It consists of two segments, each of which adapts from the red curve to the green curve individually.

well for videos corrupted by underexposure, overexposure, or mixture of them. Thus, a piecewise mapping function is needed to treat these two kinds of ill-exposed pixels differently. As shown in Figure 7, we divide our mapping function into low and high segments according to a threshold β , and each segment adapts its curvature individually. In order to get a suitable β , we introduce two threshold values, Dark and Bright; $[0, \text{Dark}]$ denotes the dark range, and $[\text{Bright}, 1]$ denotes the bright range. According to human's perception, we set Dark and Bright to 0.1 and 0.9, respectively. Perceptively, if there are more pixels falling into dark range than those into bright range, we should use low segment more and assign β a larger value. On the other hand, if there are much more pixels falling in bright range, we should use high segment more and assign β a smaller value. A simple approach to determine β is to use pixel numbers in Dark and Bright areas. Yet, owing to our APMF is calculated before the ASTC filter, there are still some noises, and pixel numbers are not quite reasonable. Thus, we use the pyramid segmentation algorithm [13] to segment a frame into several connected regions and use the region area information to determine β . Let A_i , μ_i , and σ_i denote the area, the average intensity, and the standard deviation of intensities of the i th region, respectively. Then, we compute β by

$$\beta = \frac{\sum_{\mu_i \in [0, \text{Dark}]} A_i}{\sum_{\mu_i \in [0, \text{Dark}]} A_i + \sum_{\mu_j \in [\text{Bright}, 1]} A_j}. \quad (18)$$

If β is larger than Bright, then it is assigned to 1, and the low-segment curve will occupy the whole dynamic range; if β is lower than Dark, then it is assigned to 0, and the high-segment curve will occupy the whole dynamic range. If there

are no regions with average intensities falling into either dark or bright range, then β is assigned to the default value 0.5.

With division of intensity range, the tone mapping function can be designed separately for low and high segments. Considering human perception responses, Bennett and McMillan [21] proposed a logarithmic mapping function, which well deals with underexposed videos. We incorporate their function to our adaptive piecewise mapping function (APMF) in underexposed areas but extended the function to also deal with overexposed areas as follows:

$$m(\psi_1, \psi_2, x) = \begin{cases} m_1(x, \psi_1, \lambda_1), & x \in [0, \beta] \\ m_2(x, \psi_2, \lambda_2), & x \in (\beta, 1] \end{cases}$$

$$m_1(x, \psi_1, \lambda_1) = \begin{cases} \beta \log \left(\frac{x(\psi_1 - 1)}{\beta} + 1 \right) / \log \psi_1 & \text{if } \lambda_1 > 1, \\ x & \text{if } \lambda_1 = 1, \\ \beta - \beta \log \left(\psi_1 - \frac{(\psi_1 - 1)x}{\beta} \right) / \log \psi_1 & \text{if } \lambda_1 < 1, \end{cases}$$

$$m_2(x, \psi_2, \lambda_2) = \begin{cases} \beta + (1 - \beta) \log \left(\frac{(x - \beta)(\psi_2 - 1)}{1 - \beta} + 1 \right) / \log \psi_2 & \text{if } \lambda_2 > 1, \\ x & \text{if } \lambda_2 = 1, \\ 1 - (1 - \beta) \log \left(\psi_2 - \frac{(\psi_2 - 1)(x - \beta)}{1 - \beta} \right) / \log \psi_2 & \text{if } \lambda_2 < 1, \end{cases} \quad (19)$$

where ψ_1 and ψ_2 are parameters controlling the curvatures of low and high segments, respectively. λ_1 and λ_2 are gain factors of intensities Dark and Bright, respectively, which is defined the same as λ in (15), that is, the proportion between the new intensity and the original one. λ_1 and λ_2 are precomputed before getting the mapping function and control the selection of curves between the red and the green in Figure 7. This mapping function avoids sharp slope near the origin, and thus well preserves details [21].

6.2. Automatic parameters selection

Although we designed the APMF as (19) to deal with different situations, how to choose appropriate parameters in the function determines the tone mapping performance. Thus, we will detail the process of choosing these important parameters— λ_1 , λ_2 , ψ_1 , ψ_2 , and ψ_2 .

When certain dynamic range is enlarged, there must be some other ranges being compressed. As to an intensity range $[I_1, I_2]$, if more segmented regions fall into it, then there is probably more information in this range, and thus the contrast should be enlarged, that is enlarging the intensity range. On the other hand, if the standard deviation of regions in this range is quite large, then it is probably that the contrast is already enough and needs not to be enlarged anymore [30].

According to the above, we define the enlarged range R of $[I_1, I_2]$ as

$$R(I_1, I_2, I) = (I - (I_2 - I_1))e^{-\sum_{\mu_i \in [I_1, I_2]} ((N(\sigma_i))/N(A_i))}, \quad (20)$$

where N is the normalization operator (divided by the maximum), and I is the maximum range which can be stretched to. In other words, $(I - (I_2 - I_1))$ denotes the maximum enlarging range, and the exponential factor controls the enlarging scale. It should be noticed that the segmented regions with too small standard deviation should be disregarded in (20) because they probably correspond to the backgrounds or monochromatic boards in the image and should not be enhanced anymore.

We take the low segment curve in Figure 7 as an example. If $[0, \text{Dark}]$ is enlarged, the red curve should be adopted, and Dark is extended to $\text{Dark} + l_1$. The maximum of l_1 is $\beta - (\text{Dark} - 0)$, and thus l_1 can be represented as $R(0, \text{Dark}, \beta)$. Similarly, if $[\text{Dark}, \beta]$ is enlarged, the green curve should be adopted, and Dark is compressed to $\text{Dark} - l_2$, in which l_2 is represented as $R(\text{Dark}, \beta, \beta)$. Therefore, considering both parts, we make the new mapping intensity of Dark as $\text{Dark} + l_1 - l_2$. Then λ_1 is $(\text{Dark} + l_1 - l_2)/\text{Dark}$, and ψ_1 can be computed by solving the following equation:

$$m_1(\text{Dark}, \psi_1, \lambda_1) = \text{Dark} + R(0, \text{Dark}, \beta) - R(\text{Dark}, \beta, \beta), \quad (21)$$

λ_2 and ψ_2 can be got similarly. Thus, all the parameters in (19) are determined.

As mentioned in Section 2, in order to better deal with details as well as avoiding ringing artifacts, we first separate an image into large scale parts and details using ROAD bilateral filter owing to its ability of well preserving fine features [26], and then enhance the large scale parts with function $m(\psi_1, \psi_2, x)$, while enhancing details with a less curved function $m(\psi_1 \times e^{-N(\sigma_L)}, \psi_2 e^{-N(\sigma_H)}, x)$. σ_L and σ_H correspond to the intensity standard deviations of all regions falling into $[0, \beta]$ and $(\beta, 1]$, respectively. The larger the standard deviation is, the more linear the mapping function for the details is.

APMF can also avoid introducing washed-out artifacts, that is, over enhancing images with homochromatic backgrounds. Figure 8(a) shows an image of moon with black background. The histogram equalization result exhibits a washed-out appearance shown in Figure 8(b), for the reason that the background corresponds to the largest component in histogram and causes the whole picture enhanced too much [12]. Figure 8(c) shows the result of the most popular image processing software, Photoshop, using its “Auto Contrast” function [31]. The disappointing appearance comes from its disregarding the first 0.5% of the range of white and black pixels, which leads to loss of information in the clipped ranges. Figure 8(d) shows the APMF result, and we can see that the craters in the central of image are quite clear.

6.3. Temporal filtering of APMF

APMF is formed based on the statistical information of each frame separately, and differences contained in the

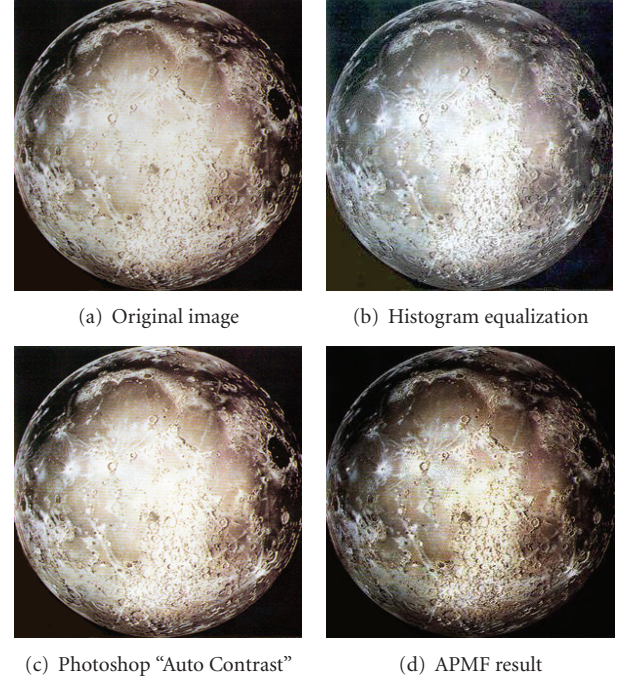


FIGURE 8: Comparison of different contrast enhancement approaches.

successive frames may result in disturbing flicker. Small difference means that the scene of video is very smooth and the flicker can be reduced by smoothing the mapping functions. Large difference probably means that a shot cut occurring and the current mapping function should be replaced by a new one. Since APMF is determined by three values— β , $m(\psi_1, \psi_2, \text{Dark})$, and $m(\psi_1, \psi_2, \text{Bright})$, we define the function difference as

$$\text{Diff} = \Delta\beta + \Delta m(\psi_1, \psi_2, \text{Dark}) + \Delta m(\psi_1, \psi_2, \text{Bright}), \quad (22)$$

where Δ is the difference operator. If Diff of successive frames is lower than a threshold, then we smooth β , $m(\psi_1, \psi_2, \text{Dark})$, and $m(\psi_1, \psi_2, \text{Bright})$ in the APMF of current frame by averaging corresponding values in neighboring $(2m + 1)$ frames. Otherwise, we just adopt the new APMF. In our experiments, m is fixed to 5 and the threshold is 30.

7. EXPERIMENTS

To demonstrate the effectiveness of the proposed video enhancement framework, we have applied it to a broad variety of low-quality videos, including corrupted by mixed Gaussian and impulse noise, underexposed and overexposed video sequences. Although it is difficult to obtain the ground truth comparison for video enhancement, it can be clearly seen from the processed results that our framework is superior to the other existing methods.

First, we compare performances of our video enhancement system with ASTA system. Since ASTA can only work for underexposed videos, we only do the comparison on such



FIGURE 9: Underexposed video results. (a) Test video added by impulse ($p = 10\%$) noise. (b) Result of histogram equalization. (c) Result of ASTA filter followed by histogram equalization. (d) Result of ASTA system. (e) Result of our system.

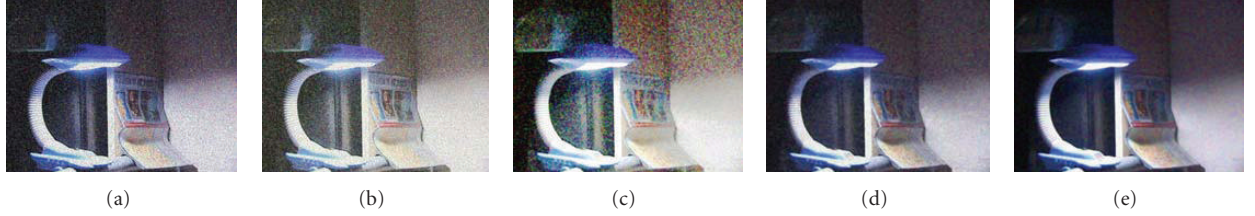


FIGURE 10: Overexposed video results. (a) Test video added by mixed impulse ($p = 10\%$) and Gaussian ($\sigma = 10$) noise. (b) Result of histogram equalization. (c) Result of AML3D filter followed by histogram equalization. (d) Result of AML3D filter followed by our APMF. (e) Result of our system.

videos. In addition, we also make comparisons with other two most common 3-dimensional median filters—P3D [32] filter and AML3D [33] filters followed by histogram equalization and our APMF. The results are shown in Figures 9, 10, and 11, which are experiments on underexposed video, overexposed video, and video with under- and over-exposed regions. Since underexposed regions are assumed owning zero-mean Gaussian noise [21], we only add uniformly distributed impulse noise to such videos as shown in Figures 9(a), 11(a), but add mixed noise to over-exposed video as shown in Figure 10(b).

From all picture (b), (c) of Figures 9, 10, and 11, all of which are enhanced by the popular contrast enhancement method-histogram equalization, we can see that no matter whether the noises are filtered in advance (all Figure (c)) or not (all Figure (b)), the output videos are always unacceptable, since the noises are over-enhanced in the equalization process. While our APMF considers the intensity standard deviations and treat large scale parts and details differently. From Figures 10(c), 10(d), 11(c), and 11(d), we can see that our APMF produces much better outputs than histogram equalization after the same filtering process. Our APMF greatly enhances the video as well as suppressing mixed noises. In addition, our APMF produces desirable outputs in all underexposed, overexposed, and mixed ill-exposed videos, owing to its ability of adaptively adjusting the mapping functions according to different videos.

As to noise filtering, our ASTC filter also outperforms other approaches. Although the ASTA system works well on videos with Gaussian noises [21], it fails to deal with videos with mixed noises as shown in Figure 9(d). We can see great impulse noise pixels all over the image. This is because ASTA is formed by combining the spatial and temporal bilateral filters, which take the impulse noise pixels as “temporal

edges” and leave them untouched. In addition, AML3D filter and P3D filter, which are two kinds of improved spatio-temporal median filters, produce grainy results as shown in the bright wall regions in Figure 10(d) as well as the dark regions in Figure 11(d). In contrast, our system produces more pleasing outputs as shown in Figure 10(e) and well preserves details that are hardly visible in the original videos such as the car in Figure 9(e) and the telephone in Figure 11(e). The reason is that our noise filter is based on the combination of a good impulse detector and the classical bilateral filter; the former well deals with large outliers, and the latter effectively smoothes small amplitude noises. In general, the results indicate the robustness and effectiveness of our video enhancement system in different kinds of videos with mixed noises.

8. CONCLUSIONS

In this paper, we have presented a universal video enhancement system, which is able to greatly suppress the most two common noises—Gaussian and impulse noises as well as significantly enhance video contrast. We introduce a novel local image statistic—neighborhood connective value (NCV) to improve impulse noise detection performance to a great extent. Then, we incorporate it into the bilateral filter framework to form an adaptive spatio-temporal connective (ASTC) filter to reduce mixed noises. ASTC filter adapts from a temporal filter to a spatial one based on noise level and local motion characteristics, and thus assure its robustness for different videos. Furthermore, we build an adaptive piecewise mapping function (APMF) to automatically enhance video contrast using statistical information of frame segmentation results, which provide more 2-D spatial information than the histogram statistics. We conducted a simulation experiment

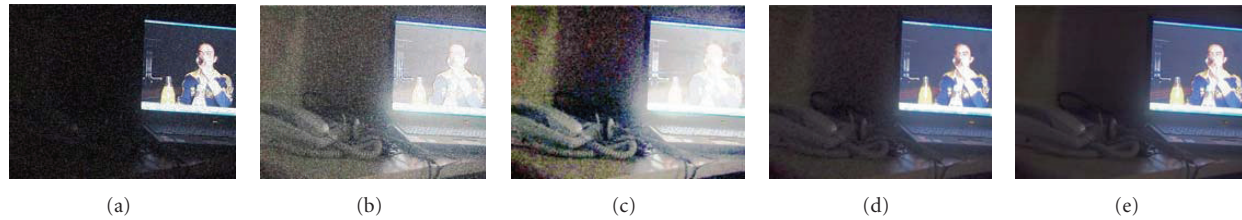


FIGURE 11: Results of video with under- and over-exposed regions. (a) Test video added by impulse ($p = 10\%$) noise. (b) Result of histogram equalization. (c) Result of P3D filter followed by histogram equalization. (d) Result of P3D filter followed by our APME. (e) Result of our system.

on three representative images, and an extensive experiment on several videos, which are underexposed, overexposed, or both of them. Both the objective and subjective evaluations indicated the effectiveness of our system.

Limitations remain in our system, however. First, our system assumes that impulse noise pixels are always closely connected with fewer neighboring pixels than signal pixels, so it will fail to remove large blotches (i.e., distorted region larger than four pixels) for film restoration. Secondly, our implementation is very slow since it includes multiple nonlinear filtering steps and computation of NCVs. The current processing of one 720×576 frame takes about one minute. Extending our approach to detect large blotches and improving its performance are our future work. Furthermore, we will pay attention to enhance video regions differently according to human's attention model.

ACKNOWLEDGMENTS

This work was supported by the National High-Tech Research and Development Plan (863) of China under Grant no. 2006AA01Z118, National Basic Research Program (973) of China under Grant no. 2006CB303103, and National Natural Science Foundation of China under Grant no. 60573167.

REFERENCES

- [1] S. Peng and L. Lucke, "Multi-level adaptive fuzzy filter for mixed noise removal," in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS '95)*, vol. 2, pp. 1524–1527, Seattle, Wash, USA, April-May 1995.
- [2] R. Garnett, T. Huegerich, C. Chui, and W. He, "A universal noise removal algorithm with an impulse detector," *IEEE Transactions on Image Processing*, vol. 14, no. 11, pp. 1747–1754, 2005.
- [3] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proceedings of the 6th IEEE International Conference on Computer Vision (ICCV '98)*, pp. 839–846, Bombay, India, January 1998.
- [4] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639, 1990.
- [5] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *IEEE Transactions on Image Processing*, vol. 12, no. 11, pp. 1338–1351, 2003.
- [6] S. Roth and M. J. Black, "Fields of experts: a framework for learning image priors," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, vol. 2, pp. 860–867, San Diego, Calif, USA, June 2005.
- [7] G. Pok, J.-C. Liu, and A. S. Nair, "Selective removal of impulse noise based on homogeneity level information," *IEEE Transactions on Image Processing*, vol. 12, no. 1, pp. 85–92, 2003.
- [8] V. Kober, M. Mozerov, and J. Alvarez-Borrego, "Nonlinear filters with spatially connected neighborhoods," *Optical Engineering*, vol. 40, no. 6, pp. 971–983, 2001.
- [9] W.-Y. Han and J.-C. Lin, "Minimum-maximum exclusive mean (MMEM) filter to remove impulse noise from highly corrupted images," *Electronics Letters*, vol. 33, no. 2, pp. 124–125, 1997.
- [10] J. C. Brailean, R. P. Kleihorst, S. Efstratiadis, A. K. Katsaggelos, and R. L. Legendijk, "Noise reduction filters for dynamic image sequences: a review," *Proceedings of the IEEE*, vol. 83, no. 9, pp. 1272–1292, 1995.
- [11] J. K. Aggarwal and N. Nandhakumar, "On the computation of motion from sequences of images—a review," *Proceedings of the IEEE*, vol. 76, no. 8, pp. 917–935, 1988.
- [12] Z. Chen, B. R. Abidi, D. L. Page, and M. A. Abidi, "Gray-level grouping (GLG): an automatic method for optimized image contrast enhancement—part I: the basic method," *IEEE Transactions on Image Processing*, vol. 15, no. 8, pp. 2290–2302, 2006.
- [13] M. Bister, J. Cornelis, and A. Rosenfeld, "A critical view of pyramid segmentation algorithms," *Pattern Recognition Letters*, vol. 11, no. 9, pp. 605–617, 1990.
- [14] R. van den Boomgaard and J. van de Weijer, "On the equivalence of local-mode finding, robust estimation and mean-shift analysis as used in early vision tasks," in *Proceedings of the 16th International Conference on Pattern Recognition (ICPR '02)*, vol. 3, pp. 927–930, Quebec, Canada, August 2002.
- [15] J. J. Francis and G. D. Jager, "The bilateral median filter," in *Proceedings of the 14th Annual Symposium of the Pattern Recognition Association of South Africa (PRASA '03)*, Langebaan, South Africa, November 2003.
- [16] D. Barash, "A fundamental relationship between bilateral filtering, adaptive smoothing, and the nonlinear diffusion equation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 6, pp. 844–847, 2002.
- [17] R. C. Hardie and K. E. Barner, "Rank conditioned rank selection filters for signal restoration," *IEEE Transactions on Image Processing*, vol. 3, no. 2, pp. 192–206, 1994.
- [18] E. Abreu, M. Lightstone, S. K. Mitra, and K. Arakawa, "A new efficient approach for the removal of impulse noise

- from highly corrupted images,” *IEEE Transactions on Image Processing*, vol. 5, no. 6, pp. 1012–1025, 1996.
- [19] S. H. Lee and M. G. Kang, “Spatio-temporal video filtering algorithm based on 3-D anisotropic diffusion equation,” in *Proceedings of IEEE International Conference on Image Processing (ICIP '98)*, vol. 2, pp. 447–450, Chicago, Ill, USA, October 1998.
 - [20] K. Jostschulte, A. Amer, M. Schu, and H. Schröder, “Perception adaptive temporal TV-noise reduction using contour preserving prefilter techniques,” *IEEE Transactions on Consumer Electronics*, vol. 44, no. 3, pp. 1091–1096, 1998.
 - [21] E. P. Bennett and L. McMillan, “Video enhancement using per-pixel virtual exposures,” *ACM Transactions on Graphics*, vol. 24, no. 3, pp. 845–852, 2005.
 - [22] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Prentice-Hall, Englewood Cliffs, NJ, USA, 2nd edition, 2002.
 - [23] K. H. Goh, Y. Huang, and L. Hui, “Automatic video contrast enhancement,” in *Proceedings of IEEE International Symposium on Consumer Electronics (ISCE '04)*, pp. 359–364, Reading, UK, September 2004.
 - [24] A. Polesel, G. Ramponi, and V. J. Mathews, “Image enhancement via adaptive unsharp masking,” *IEEE Transactions on Image Processing*, vol. 9, no. 3, pp. 505–510, 2000.
 - [25] F. Durand and J. Dorsey, “Fast bilateral filtering for the display of high-dynamic-range images,” *ACM Transactions on Graphics*, vol. 21, no. 3, pp. 257–266, 2002.
 - [26] E. P. Bennett and L. McMillan, “Fine feature preservation for HDR tone mapping,” in *Proceedings of the 33rd International Conference and Exhibition on Computer Graphics and Interactive Techniques (SIGGRAPH '06)*, Boston, Mass, USA, July-August 2006.
 - [27] S. Schulte, M. Nachtegaal, V. De Witte, D. Van der Weken, and E. E. Kerre, “A fuzzy impulse noise detection and reduction method,” *IEEE Transactions on Image Processing*, vol. 15, no. 5, pp. 1153–1162, 2006.
 - [28] E. W. Dijkstra, “A note on two problems in connexion with graphs,” *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.
 - [29] G. Zhu, C. Xu, Q. Huang, W. Gao, and L. Xing, “Player action recognition in broadcast tennis video with applications to semantic analysis of sports game,” in *Proceedings of the 14th Annual ACM International Conference on Multimedia*, pp. 431–440, Santa Barbara, Calif, USA, October 2006.
 - [30] D.-C. Chang and W.-R. Wu, “Image contrast enhancement based on a histogram transformation of local standard deviation,” *IEEE Transactions on Medical Imaging*, vol. 17, no. 4, pp. 518–531, 1998.
 - [31] Adobe Systems, Inc., Adobe Magazine May-June 2000, <http://www.adobe.com/products/adobemag/archive/pdfs/0005qaps.pdf>.
 - [32] M. B. Alp and Y. Neuvo, “3-dimensional median filters for image sequence processing,” in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '91)*, vol. 4, pp. 2917–2920, Toronto, Canada, April 1991.
 - [33] S. Jackson and A. Savakis, “Adaptive multilevel median filtering of image sequences,” in *Proceedings of IEEE International Conference on Image Processing (ICIP '05)*, vol. 3, pp. 545–548, Genova, Italy, September 2005.